2009 MEMOCODE Co-Design Contest

Forrest Brewer
University of California at Santa Barbara
forrest@ece.ucsb.edu

James C. Hoe
Carnegie Mellon University
jhoe@ece.cmu.edu

Abstract

The 2009 MEMOCODE Co-Design Contest is the third in the series of annual design contests organized by the MEMOCODE Conference. Contestants have one month to create the best performing design solution to a posted design challenge. The contest is open to all interested participants, and the contest rules are designed to not exclude or favor any one design methodology or platform. The goal of the contest is to invite developers of tools and platforms to showcase their technology in a leveled competition and to encourage hands-on design activities in the fields of interest of the MEMOCODE Conference. Please see http://www.memocode-conference.com *for current information about this contest.*

1. Introduction

The 2009 MEMOCODE Co-Design Contest is the third in the series of annual contests organized by the MEMOCODE Conference. As in the previous years, the contest follows an open-format that places very little restrictions on who can participate and what methodologies and platforms may be used to enter.

The two major requirements are: 1. The submitted entry must be an actual demonstratable design (i.e., simulated designs and results are not acceptable) and 2. The entry must be completed within the one-month period after the design challenge is posted. Entries compete in two prize categories: absolute performance and normalized performance. The absolute performance contest is an anything-goes competition for the highest performance implementation; the normalized performance contest takes platform capability into consideration to reward efficiency and to enable contestants with any level of resources to compete on an even footing.

The contest is intended to promote three major goals: First, the contest aims to encourage interest in hands-on hardware/software co-design activities in both academic and industrial settings. Second, the contest attempts to provide an open, unbiased forum where academic and industry tool developers can showcase the advantages and issues in their design methodologies or platforms. Lastly, the design challenge and the wide variety of solutions collected over the years (most of which are available in open-source forms on the contest website) serve as openly available best-effort benchmarks to be used by our community for any purpose.

2. Contest Rules

The 2009 contest follows the same rules as established for the 2008 contest. Contestants have one month from the time when the design challenge is posted to complete and submit a solution. The solution must work correctly to be considered for awards.

Eligibility. The contest is open to industry and academic participation. A team may include both industry and academic members. There is no limit on the number of members of a team. There is no limit on the number of teams per institution. However, each person may participate on only one team.

Tools and Platforms. Contestants may use any hardware and software design methodologies at their disposal; formal methods are encouraged but not required. The contestants are also allowed to make use of existing IPs available to them. The contestants may use any development platforms without limit on the number of processors and FPGA devices, except the platform should have at least 512 MBytes of memory. This open contest format is designed to encourage the contestants to bring their best and most familiar technology to the contest. The contest does designate the Xilinx XUP2VP development board as the contest's reference platform by providing a reference software-only solution for that board as a part of the design challenge specification.

Metrics and Judging. For 2009, an entry is evaluated for both absolute performance and normalized performance. In addition, a subjective element of judging is based on the elegance of the solutions as determined by a panel of three judges. This year, in

978-1-4244-4807-4/09/$25.00 ©2009 IEEE

addition to the two organizers, Dr. Kees Vissers (Xilinx) served as the third member of the panel. For each category of the performance competition, the entries are ranked overall by the value ($Rank_{performance}$ + $Rank_{elegance}$). In the case of a tie, $Rank_{performance}$ is the tie-breaker.

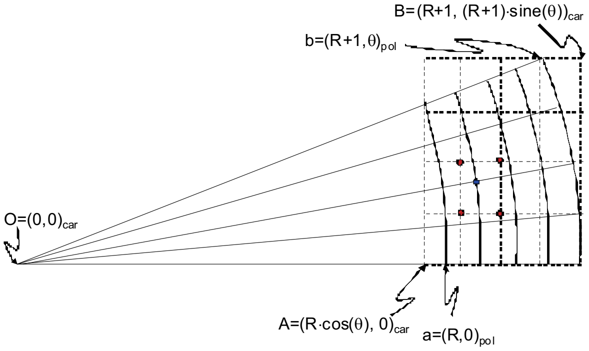
The objective performance metric used in ranking is the *geometric* average speedup (over a prescribed set of test inputs) of a contestant's implementation relative to a provided reference software-only implementation. For the absolute performance category, speedup is computed using the wall-clock execution time. For the normalized performance category, speedup is computed using a normalized execution time (discussed next).

Performance Normalization. The contest's normalization rule derates the achieved speedup of an entry according to a formula that takes into consideration the number and the performance of the processors employed and the total reconfigurable hardware capacity employed. In the normalized performance category, teams with access to a wide range of resources can compete fairly in creating the most "efficient" implementation. The organizers are aware that the prescribed normalization rule can not be perfectly fair in all aspects—for example, memory capacity and bandwidth is not currently a factor in the performance normalization. This and other inadequacies are explicitly acknowledged in the contest rules.

New to this year's contest were submissions based on GPUs (graphics processing units). During the judging period, the judges realized that the current normalization rule is grossly inadequate to address the performance characteristics of the GPU-based entries. As a result, the judges decided not to recognize the normalized performance results from the two GPU-based entries. (The two GPU-based entries would not have been competitive in the normalized performance contest by even very generous estimations.) Next year's contest organizers will address this issue to enable a fair normalization across a wider range of platforms.

3. Design Challenge

The 2009 contestants were tasked to implement a system to compute the values over an N×N grid specified in polar coordinates by interpolating the values from an enclosing N×N grid specified in the Cartesian coordinates. This "made-up" problem is a greatly simplified version of the interpolation problem normally found in practice. The problem was designed to be accessible to contestants regardless of their domain knowledge and to emphasize the proper handling of concurrency and data locality in the solutions. We briefly describe the design challenge below; please see the contest website for the complete specification.



In the figure above, the region C is defined by a bounding box in Cartesian coordinates, with points $A=(R\cdot\cos(\theta),0)_{Cartesian}$ and $B=(R+1, (R+1)\cdot\text{sine}(\theta))_{Cartesian}$ on opposite corners, where $10 \leq R \leq 100$, $(\pi/256) \leq \theta \leq (\pi/4)$. Region C is evenly spanned by an NxN grid where N is an integer between $10 \leq N \leq 1000$. The values associated with the grid points are stored in a two-dimensional array CART. Similarly, a region P is defined by a bounding box in polar coordinates, with points $a=(R,0)_{polar}$ and $b=(R+1, \theta)_{polar}$ on opposite corners. Region P is fully-enclosed by region C. Region P is also evenly spanned by an NxN grid, and the values associated with the grid points are stored in a two-dimensional array POL.

The contestants have one month to implement a system that computes, for valid values of N, R and θ, the contents of the output array POL given the input array CART. To keep the problem accessible to all levels of contestants, the interpolated value associated with a grid point in P is taken to be the simple average of the 4 enclosing grid points in C. The implementation must be able to handle all valid values of N, R and θ as parameters without recompilation or resynthesis. The complete design specification further specifies the input/output data formats, the required accuracy and the required initial and final conditions. The design challenge specification includes a reference software-only solution for the Xilinx XUP2VP development board.

4. Contest Results

The design challenge was posted on midnight, March 1st. This year, twenty-two teams from around the world registered for the contest. The starters included teams from (in order of registration) Old

Dominion University, UVA, MIT, Bradley University, Royal Institute of Technology (Stockholm), iicdesign.com, Politecnico di Milan, Virginia Tech, University of Idaho, IIT Madras, two teams from TU-Graz, UC Davis, two teams from Iowa State University. Ten teams were still active in the final week of the contest. Ultimately, five teams submitted finished solutions by the deadline on March 31st. The five teams and their results are summarized in the order of submission below.

Barracuda (Iowa State University): A. Veerendra, J.-N. Tioh, J. Rilling, L. Seshagiri, M. Steffen

- **Platform:** NVIDIA Tesla T10
- **Development:** NVIDIA CUDA
- **Speedup**: absolute=24371 (2nd place);
  normalized=NA

CA$HE MON3Y (Old Dominion University): W.H. Edwards, N. Gosnel, Jr., A. Lewis

- **Platform:** XUPV2P, software only
- **Development:** ISE/EDK
- **Speedup:** absolute=2.4 (5th place);
  normalized=2.4 (3rd place);

Team MIT (MIT): A. Agarwal, N. Dave, K. Fleming, A. Khan, M. King, M. Ng, M. Vijayaraghavan

- **Platform:** XUPV2P
- **Development**: Bluespec, ISE/EDK
- **Speedup**: absolute=3381 (3rd place);
  normalized=3381 (1st place);

TeleTitanium (independent): D.L. Rosenband and T. Rosenband

- **Platform:** AMD Athlon 64 X2 Dual Core 4200+ with NVIDIA GTX 285 GPGPU
- **Development:** gcc and NVIDIA CUDA
- **Speedup**: **absolute=53064 (1st place);**
  normalized=NA;

Uhrturm (IAIPC, Graz University of Technology): E. Wenger and P. Rouschal

- **Platform:** XUPV2P
- **Development:** ISE/EDK and Matlab for analysis
- **Speedup:** absolute=462 (4th place);
  normalized=462 (2nd place);

The standouts in this year's absolute performance contest are the two GPU-based entries. As noted earlier, due to the fault of the normalization rules, we could not attribute an appropriate normalized speedup to the two GPU-based entries. (It should be noted however that the judges did ascertain that the two GPU-based entries would not have been competitive in the normalized performance contest by even very generous estimations.)

After a month of deliberation, the 2009 judges—Kees Vissers (Xilinx); Forrest Brewer (UC Santa Barbara); and James C. Hoe (CMU)—unanimously arrived at this year's two winning designs. The winner of the Absolute Performance Prize is Team TeleTitanium. The winner of the Normalized Performance Prize is Team MIT. The functionality and performance achieved by the two winning teams have been verified the judges. The two winning teams will present their designs and be awarded with a $1000 cash prize at the conference. They are also invited to contribute a 4-page paper in the formal proceedings of the 2009 MEMOCODE Conference. Each team that submitted a completed design is also eligible to submit for review a 2-page abstract for the formal conference proceedings.

5. Concluding Remarks

We are glad to see the third running of the MEMOCODE Co-Design Contest come to a successful and exciting conclusion. This year, we saw GPU platforms entering the field to take the Absolute Performance Prize by a wide margin. Nevertheless, craftiness of design continues to be the most important factor in winning the Normalized Performance Prize. In future years, we hope to see a larger field of contestants and a greater variety of platforms and methodologies to compete for peak performance and/or efficiency in this contest. The organizers next year will endeavor to devise a simple and yet more generally applicable normalization rule to support this vision.

In closing, we want to thank everyone that participated in this year's contest. We like to thank IEEE Council on Electronic Design and Automation (CEDA), Bluespec, and Xilinx for their support. We like to thank Dr. Kees Vissers (Xilinx) for volunteering to serve on this year's judging panel. Finally, a special acknowledgement goes to Rachata Ausavarungnirun (CMU) for preparing this year's reference implementation.